AppleScript-hjælp

Vigtige emner

Om AppleScript

Når du arbejder med Mac, vil du opdage, at der er nogle rutineopgaver, som du udfører gang på gang. Med AppleScript kan du skrive instrukser og programmer, som kan automatisere rutineopgaver.

AppleScript er et effektivt og fleksibelt instruktssprog, som er indbygget i Mac OS X. Du kan bruge AppleScript til at oprette genveje, automatisere gentagne opgaver og endda til at oprette specielle programmer, så du kan spare megen tid. Med AppleScript kan du styre programmer, der kan arbejde med instrukser.

En AppleScript-instruks består af nogle kommandoer, der ligner almindelige engelske ord og udfører handlinger på objekter. Eksempler på disse handlinger er at åbne, lukke eller vise navnet på et emne. Et objekt kan f.eks. være et program, et vindue eller en indstilling.

En instruks kan styre et eller flere programmer. Du kan kun bruge instrukser til programmer, som definerer, hvilke kommandoer der kan bruges i instrukserne. OS X leveres med flere programmer, som kan arbejde med instrukser.

Instrukser kan også styre programmer via et netværk, så du kan automatisere opgaver på flere computere.

Du har faktisk brugt AppleScript uden at vide det. Mange funktioner i systemet og i forskellige programmer er i virkeligheden AppleScript-instrukser, som kaldes fra et program. I dette hjælpesystem er det f.eks. en AppleScript-instruks, der udfører handlingen, hver gang du klikker på en henvisning, som åbner et program eller et arkiv.

BESLÆGTEDE EMNER

Hvordan fungerer AppleScript?
Om AppleScript-redigeringsværktøj
Om AppleScript-sproget
AppleScript-ressourcer på internettet
Udviklingsmiljøet AppleScript Studio

Vigtige emner

Om AppleScript-redigeringsværktøj

AppleScript-redigeringsværktøj er et brugervenligt instruksprogram, som kan læse, skrive, optage og arkivere AppleScript-instrukser.

Her følger nogle af funktionerne i AppleScript-redigeringsværktøj:

- Sikrer en ensartet og mere brugervenlig grænseflade
- Giver dig mulighed for at finde og erstatte tekst i instrukser
- Ombryder automatisk lange instruksafsnit
- Giver dig mulighed for at fortryde flere trin
- Indeholder en genvejsmenu, som gør det hurtigere at skrive instrukser

- Understøtter programtjenester
- Giver dig adgang til programinstrukser vha. Bibliotekspaletten
- Indeholder en instruksassistent, der hjælper dig med at færdiggøre kommandoer
- Indeholder resultatoversigt og logoversigt for begivenheder, så du kan undersøge, hvad der sker i en instruks

BESLÆGTEDE EMNER

Om AppleScript-sproget
Udviklingsmiljøet AppleScript Studio

Vigtige emner

Opret AppleScript-instrukser

Du behøver ikke at vide noget om programmering, før du kan nyde godt af AppleScript. Du behøver faktisk ikke en gang at skrive et eneste ord i en instruks. OS X indeholder flere hundrede praktiske AppleScripts-instrukser, som du kan begynde at bruge, og der findes mange tusind på internettet.

Hvis du vil fordybe dig endnu mere, kan du tilpasse eksisterende instrukser efter dine behov ved at foretage mindre ændringer.

Du får selvfølgelig størst udbytte af at oprette dine egne instrukser. Hvis du lærer at skrive instrukser, kan du oprette praktiske værktøjer og tidsbesparende automatiske funktioner, så du kan øge produktiviteten.

Du kan ændre eller oprette dine egne instrukser vha. AppleScript-redigeringsværktøj, som indeholder et enkelt miljø, hvor du kan skrive instrukser. Alle eksemplerne i AppleScript-hjælp er fremstillet vha. AppleScript-redigeringsværktøj.

- Åbn AppleScript-redigeringsværktøj.

  Skriv følgende tekst i vinduet med AppleScript-redigeringsværktøj.

  ```
  say "Velkommen til AppleScript!"
  ```

  Klik på start.

BESLÆGTEDE EMNER

Om AppleScript-redigeringsværktøj
Udviklingsmiljøet AppleScript Studio

Vigtige emner

Hvordan fungerer AppleScript?

AppleScript arbejder ved at sende beskeder, som kaldes “Apple events”. Når du afvikler en instruks, sender AppleScript nogle Apple events til systemet eller til de programmer, som skal udføre kommandoerne.

I mange tilfælde returnerer programmerne eller systemet et resultat til AppleScript. Hvis en instruks f.eks. sender en anmodning til et program om at få navnet på et objekt, returnerer programmet navnet. Hvis kommandoen ikke kan udføres, viser programmet eller AppleScript måske en fejlmeddelelse. Du kan se de resultater, der returneres, i AppleScript-redigeringsværktøj.

Generelt behøver du ikke at bekymre dig om, hvordan Apple Events fungerer. AppleScript udfører dem i baggrunden. Du skal blot skrive og afvikle dine instrukser.

BESLÆGTEDE EMNER

Om AppleScript
Om AppleScript-sproget

Vigtige emner

Om miniprogrammer (applets)

En instruks, der er arkiveret som et program, kaldes et AppleScript-program, et miniprogram eller "applet". Miniprogrammer afvikles automatisk, når de åbnes, og udfører kommandoerne i instruksen. Miniprogrammer kan enten være små funktioner, der kun indeholder nogle få kommandoer, eller komplekse programmer, der indeholder tusindvis af linjer.

Miniprogrammer kan arkiveres, så de kan redigeres eller som "run-only". Et miniprogram, som kan redigeres, indeholder stadig teksten i instruksen, så den altid kan redigeres og kompileres. Du redigerer eller ændrer et miniprogram ved at trække det til symbolet for AppleScript-redigeringsværktøj. Hvis du arkiverer et miniprogram som "run-only", fjernes den tekst, der kan redigeres, og arkivet kan ikke længere redigeres eller kompileres igen.

Du redigerer et miniprogram ved at åbne det i AppleScript-redigeringsværktøj eller et andet program til redigering af AppleScript-instrukser. Miniprogrammer kræver en såkaldt “on run”-rutine, som begynder at afvikle instruksen, når miniprogrammet åbnes.

- Dette eksempel viser en såkaldt “on run”-rutine.

```
on run
        --første kommando
        --anden kommando
end run
```

BESLÆGTEDE EMNER

Om "droplets"
AppleScript-arkivtyper

Vigtige emner

Om kompilerede instrukser

En kompileret instruks er et AppleScript-arkiv, som kan redigeres og læses i AppleScript-redigeringsværktøj eller et andet program til redigering af AppleScript-instrukser.

Når du åbner en kompileret instruks, vises den i AppleScript-redigeringsværktøj, men den udfører ingen af kommandoerne automatisk. Derfor kan du foretage ændringer i instruksen, uden at den afvikles. Derefter kan du kompilere instruksen, hvorved al teksten i arkivet læses og konverteres til AppleScript-kode. Den kompilerede instruks kan afvikles. En kompileret instruks har arkivendelsen .scpt.

Når du arkiverer arkivet som en kompileret instruks, bevares teksten i arkivet sammen med koden. Derfor kan du altid redigere den kompilerede instruks igen.

BESLÆGTEDE EMNER

Om miniprogrammer (applets)
Om "droplets"
AppleScript-arkivtyper

Vigtige emner

Om droplets

En såkaldt "droplet" er en AppleScript-instruks, som afvikles, når du trækker arkiver eller mapper til dens symbol. Instruksen udfører handlinger med arkiverne eller mapperne.

På samme måde som miniprogrammer (applets) kan en droplet arkiveres, så den kan redigeres eller som "run-only". En droplet, som kan redigeres, indeholder stadig teksten i instruksen, så den altid kan redigeres og kompileres igen. Du redigerer eller ændrer en droplet ved at trække den til symbolet for Script Editor. Hvis du arkiverer et miniprogram som "run-only", fjernes den tekst, der kan redigeres, og arkivet kan ikke længere redigeres eller kompileres igen.

Droplets kræver en såkaldt “on open”-rutine, som begynder at afvikle instruksen, når en droplet åbnes. Denne “on open”-rutine har en parameter, som indeholder en liste over de emner, der trækkes til den pågældende droplet. En del af droplet-instruksen undersøger listen, og en anden del reagerer på emnerne. Medmindre en droplet er designet, så den kun reagerer på et enkelt emne, skulle den inkludere en gentagelsessløjfe, som gennemgår emnerne på listen og udfører handlinger på hvert enkelt emne efter tur.

- Her er et eksempel på en droplet-rutine med “on open” og parameterlisten.

```
on open (theItemsDropped)
        --første kommando
        --anden kommando
end open
```

BESLÆGTEDE EMNER

Om miniprogrammer (applets)
Om kompilerede instrukser
AppleScript-arkivtyper

Vigtige emner

Brug AppleScript-ordbøger

Du kan se, hvilke kommandoer og objekter du kan bruge til et program, ved at åbne ordbogen til det pågældende program. En ordbog indeholder et sæt termer, der kan læses af et program.

Du kan se et programs ordbog i AppleScript-redigeringsværktøj ved at vælge programmet i vinduet Bibliotek eller ved at trække programmets symbol til symbolet for AppleScript-redigeringsværktøj.

Termerne i en ordbog er organiseret i grupper af beslægtede termer kaldet “suites”. Kommandoer og objekter står med fed skrift.

1. Vælg Vindue > Bibliotek i AppleScript-redigeringsværktøj.

2. Dobbeltklik på programnavnet i biblioteksvinduet.
3. Vælg en pakke.

   Termerne i pakken og deres beskrivelser vises.

AppleScript-sproget består af forskellige kategorier:

- Suites: Grupper af beslægtede kommandoer. I ordbogen til AppleScript-redigeringsværktøj vises suites i den første kolonne mærket med et orange “S”.
- Kommandoer: Kræver en handling eller et resultat. Kommandoer, der også omtales som “begivenheder”, vises i den anden kolonne, som har et blåt “C” (omtales også som en begivenhed).
- Klasser: Objekter, der kan arbejde med instrukser. Beskrivelser af klasser viser de kendetegn og elementer, der findes i en klasse. Klasser vises i den anden kolonne, som har et lilla "C".
- Kendetegn: Objekter, som indeholder værdier. Kendetegn vises i den tredje kolonne, som har et lilla "P".
- Elementer: Objekter i andre objekter. Et afsnit kan f.eks. indeholde mange ord. Elementer kan vises i den tredje kolonne, som har et orange "E".

BESLÆGTEDE EMNER

Definitioner af klasser
Definitioner af kommander
Brug AppleScript til at betjene programmer

ÅBN

AppleScript Language Guide: Commands Reference
AppleScript Language Guide: Class Reference

Vigtige emner

Løs problemer

Hvis der opstår et problem under brugen af AppleScript, kan du udforske følgende emner for at finde oplysninger, der kan hjælpe dig med at løse problemet.

Hvis der vises en fejl, når en bruger klikker på Annuller i dialogen Choose From List

Hvis mappekommandoen afvikles på et forkert tidspunkt

Hvis instruksen ikke kan finde en disk

Grundlæggende kommandoer

Definitioner af klasser

En definition af en klasse beskriver et enkelt instruksobjekts egenskaber og elementer. En klasse kan sammenlignes med et navneord i en sætning, og klassedefinitionen svarer til definitionen af navneordet i en ordbog.

En klassedefinition i en AppleScript-ordbog begynder som regel med selve ordet i fed skrift efterfulgt af en kort beskrivelse på samme linje.

Derefter følger en liste over elementer eller objekter, som tilhører eller udgør instruksobjektet. Listen over elementer findes ikke i alle klassedefinitioner, fordi nogle instruksobjekter indeholder andre objekter, mens andre ikke gør det.

Klassedefinitionen slutter med en liste over instruksobjektets egenskaber eller kvaliteter og deres værdier.

Oplysningerne om en egenskab begynder med egenskabens navn efterfulgt af:

- En beskrivelse af det dataformat, som egenskabens værdier lagres i, f.eks. integer (heltal), point (en liste med to tal), string (tekstdata) og Boolean (sand eller falsk)
- En valgfri typeindikator, [r/o], som viser, at egenskabens værdi er skrivebeskyttet, “read-only”, (dvs. at den kun kan læses, men ikke ændres)
- En kort beskrivelse af egenskaben

Hvis et objekt har en flertalsform, kan du henvise til flere objekter.

Hvis et objekt indeholder en liste over elementer, indeholder det andre objekter. Elementets navn, som kan bruges i instruksen, står med fed skrift. Den uformaterede tekst forklarer, hvordan der skal refereres til elementet.

Objekter kan have kendetegn. Navne på kendetegn står med fed. Den uformaterede tekst viser, hvilken type værdi kendetegnet indeholder, f.eks. en streng. Kendetegn kan indeholde en fast værdi, dvs. en såkaldt "konstant", eller en række konstanter. En instruks kan hente værdien af et kendetegn og indstille værdien af et kendetegn, medmindre der står "[r/o]" (dvs. read-only) efter kendetegnets syntaks.

BESLÆGTEDE EMNER

Definitioner af kommander
Definitioner af objekter
AppleScript-kommandoer
Brug AppleScript-ordbøger

ÅBN

AppleScript Language Guide: Class Reference

Definitioner af kommander

Kommandoer er AppleScript-ord, der udfører handlinger. Kommandoer i en AppleScript-ordbog vises i den midterste kolonne med et blåt C. Definitionen af en kommando indeholder en kort beskrivelse af kommandoens funktion og syntaks. Syntaksen er den måde, kommandoen skrives på i en instruks. Syntaksen kan bestå af flere elementer.

Ord med fed skrift indgår i en kommando. Du skal skrive dem på nøjagtig samme måde som i ordbogen.

Ord med blå skrift er oplysninger, du selv skal skrive. På den måde kan du se objekter og variabler, som du selv har defineret.

Der er nogle gange valgfri dele i kommandoerne. I så fald står de i parentes. Du behøver ikke at bruge dem i instruksen.

BESLÆGTEDE EMNER

Definitioner af klasser
Brug AppleScript-ordbøger
Definitioner af begivenheder
Definitioner af objekter

Grundlæggende kommandoer

Definitioner af begivenheder

En handlingsdefinition beskriver en kommando, der peger på en handling ved et instruksobjekt. Events er de handlinger eller kommandoer, som objekter kan udføre eller deltage i.

En handlingsdefinition begynder som regel med selve ordet efterfulgt af en kort beskrivelse på samme linje.

Derefter følger en liste over evt. parametre, som bruges ved afvikling af kommandoen. Valgfri parametre står i parentes; det gør de krævede parametre ikke.

Oplysningerne om hver parameter begynder med navnet efterfulgt af en beskrivelse af det dataformat, som parameteren bruges til, og slutter med en kort beskrivelse af parameteren.

Definitionen af en kommando indeholder en kort beskrivelse af kommandoen (understreget) og dens syntaks. Syntaksen er den måde, kommandoen skrives på i en instruks. Syntaksen kan bestå af flere elementer.

Ord med fed skrift indgår i en kommando. Du skal skrive dem på nøjagtig samme måde som i ordbogen.

Ord med uformateret skrift er oplysninger, du selv skal skrive. På den måde kan du se objekter og variabler, som du selv har defineret.

Der er nogle gange valgfri dele i kommandoerne. I så fald står de i parentes. Du behøver ikke at bruge dem i instruksen.

Kommandoer kan udføres på objekter. Definitionen af et objekt indeholder syntaksen til definition af objektet, inklusive dets navn og en kort beskrivelse (understreget). Et objekts type bestemmer visse karakteristika for objekter, der tilhører den pågældende type, f.eks. de kommandoer, der kan bruges med objektet.

BESLÆGTEDE EMNER

Databasebegivenheder
Definitioner af klasser
Definitioner af kommander
Definitioner af objekter
Hvordan fungerer AppleScript?

Grundlæggende kommandoer

Definitioner af objekter

Kommandoer kan udføres på objekter. Definitionen af et objekt i en AppleScript-ordbog indeholder syntaksen til definition af objektet, inklusive dets navn og en kort beskrivelse (understreget). Et objekts type bestemmer visse karakteristika for objekter, der tilhører den pågældende type, f.eks. de kommandoer, der kan bruges med objektet.

Hvis et objekt har en flertalsform, kan du henvise til flere objekter.

Hvis et objekt indeholder en liste over elementer, indeholder det andre objekter. Elementets navn, som kan bruges i instruksen, står med fed skrift. Den uformaterede tekst forklarer, hvordan der skal refereres til elementet.

Objekter kan have kendetegn. Navne på kendetegn står med fed og blå skrift. Den uformaterede blå tekst viser, hvilken

type værdi kendetegnet indeholder, f.eks. en streng. Kendetegn kan indeholde en fast værdi, dvs. en såkaldt "konstant", eller en række konstanter. En instruks kan hente værdien af et kendetegn og indstille værdien af et kendetegn, medmindre der står “[r/o]” (dvs. read-only) efter kendetegnets syntaks.

BESLÆGTEDE EMNER

Brug AppleScript til at betjene programmer
Definitioner af kommander
Definitioner af klasser
Fortæl et objekt, hvad det skal gøre
Brug AppleScript-ordbøger

ÅBN

AppleScript Language Guide: Class Reference
AppleScript Language Guide: Kendetegn

Iif-afsnit

Et afsnit med If er en kommando, som kun udføres, hvis visse betingelser er opfyldt. Hvert afsnit med if indeholder et boolesk udtryk, der evalueres som sandt eller falsk. Kommandoerne i et afsnit med if udføres kun, hvis værdien af det booleske udtryk er sand. Afsnittet if slutter med end if. Eksempel:

```
if amountReceived > 100 then
        set memberStatus to "speciel"
end if
```

Kommandoerne i ovenstående afsnit med if udføres, hvis værdien af det booleske udtryk er sand (det modtagne beløb er større end 100). Hvis værdien af det booleske udtryk er falsk, udføres kommandoerne ikke.

Afsnittet med if kan også inkludere et afsnit med else. Denne kombination kaldes et afsnit med if-then-else. Eksempel:

```
if amountReceived > 100 then
        set memberStatus to "speciel"
else
        set memberStatus to "generel"
end if
```

Kommandoerne i første del af ovenstående afsnit udføres, hvis værdien af det booleske udtryk er sand (det modtagne beløb er større end 100). Hvis værdien af det første booleske udtryk er falsk, udføres kommandoerne i anden del af afsnittet efter else.

Du kan indlejre afsnit med if i hinanden og på den måde behandle flere betingelser.

Du kan læse mere om afsnit med if i AppleScript i AppleScript Language Guide.

BESLÆGTEDE EMNER

Rrepeat-afsnit
Definitioner af kommander
Indstil variabler i AppleScript
Konstante værdier

Rrepeat-afsnit

Du kan bruge afsnit med repeat til at udføre handlinger flere gange, afhængigt af betingelser. Et afsnit med repeat afvikler en gruppe kommandoer i en sløjfe, indtil en betingelse er opfyldt, hvorefter sløjfen sluttes. Du kan indlejre afsnit med repeat i hinanden og på den måde inddrage flere betingelser.

AppleScript indeholder en række afsnit med repeat, der kan inddrage forskellige betingelser.

repeat (forever)

Gentages konstant i en sløjfe. Brug kommandoen exit til at slutte sløjfen.

```
repeat
        -- kommando
        -- kommando
        if (x = 5) then exit repeat
end repeat
```

repeat (antal) times

Gentager det anførte antal gange.

```
repeat 5 times
        -- kommando
        -- kommando
end repeat
```

repeat while

Gentager, så længe en bestemt betingelse er sand.

```
set x to 1
repeat while (x < 5)
        -- kommando
        -- kommando
        set x to (x + 1)
end repeat
```

repeat until

Gentager, indtil en bestemt betingelse er sand.

```
set x to 1
repeat while (x = 5)
        -- kommando
```

```
        -- kommando
        set x to (x + 1)
end repeat
```

repeat with (variabel) from (startværdi) to (stopværdi)

Indstiller variablen til startværdien, hvorefter variablen øges efter hvert gennemløb af sløjfen. Gentagelsen stopper, når variablen når stopværdien.

```
repeat with x from (5) to (25)
        -- kommando
        -- kommando
end repeat
```

repeat with (variabel) in (liste)

Indstiller variablen til det første emne på listen. Med hvert gennemløb indstilles variablen til det næste emne. Gentagelsen stopper, når alle emner på listen er brugt.

```
set Minliste to {234, 342, 681, 874}
repeat with x in Minliste
        -- kommando
        -- kommando
end repeat
```

Du kan læse mere om kommandoer med repeat i AppleScript i AppleScript Language Guide.

BESLÆGTEDE EMNER

Iif-afsnit
Indstil variabler i AppleScript
Konstante værdier

ÅBN

AppleScript Language Guide - repeat

Grundlæggende kommandoer

Konstante værdier

En konstant er et reserveret AppleScript-ord, der indeholder en foruddefineret værdi. AppleScript indeholder flere konstanter, som du kan bruge til at få oplysninger om objekter, udføre sammenligninger eller udføre beregninger. Programmer og andre objekter kan definere deres egne konstanter. Konstanter defineres i definitioner af værdiklasser.

Du kan se værdien af en konstant, hvis du indsætter følgende eksempel i AppleScript-redigeringsværktøj og derefter klikker på Start. Klik på Resultat for at se konstantens værdi.

Resultatet er også en konstant. Det er en instrukskonstant, som refererer til den værdi, en kommando viser.

```
get the current date
```

Her følger nogle andre eksempler på konstanter:

- Matematiske konstanter: pi
- Booleske konstanter: true, false
- Dato- og tidskonstanter: minutes, hours, days, weeks
- Strengkonstanter: return, space, tab
- Instrukskonstanter: anything, current application, it, me, missing value, my, result

Du kan læse mere om konstanter i AppleScript i AppleScript Language Guide.

BESLÆGTEDE EMNER

Indstil variabler i AppleScript
Definitioner af objekter
Definitioner af klasser
Definitioner af begivenheder

ÅBN

AppleScript Language Guide - constants

Indstil variabler i AppleScript

En variabel er en beholder, der oprettes i en instruks og bruges til at opbevare en værdi. Du anfører et navn til variablen, også kaldet en id, når du opretter den. Når en instruks afvikles, kan du anbringe en værdi i variablen, få adgang til værdien eller ændre den. Variabler findes kun, mens instruksen afvikles. Når instruksen slutter, går værdien tabt.

- Disse eksempler viser, hvordan du anfører en entydig id og tildeler den en værdi.

```
set myVar to "153"

set x to "Kvartalsregnskaber"

set x to false
```

Du kan se, hvordan værdier lagres, hvis du indsætter disse eksempler i AppleScript-redigeringsværktøj og derefter klikker på Start. Klik på Resultat for at se den værdi, variablen indeholder.

Du kan læse mere om variabler i AppleScript i AppleScript Language Guide.

BESLÆGTEDE EMNER

Konstante værdier
Iif-afsnit
Rrepeat-afsnit

ÅBN

AppleScript Language Guide - variables

Grundlæggende kommandoer

Anfør et programs placering

Programmer i OS X er sat sammen som “pakker”. En pakke er en speciel mappe, der indeholder de mapper og arkiver, som programmet består af. Derfor kan kommandoerne “path to” og “info for” vise forskellige resultater for et program.

Hvis du bruger kommandoen “path to” til et program, slutter stien normalt med “.app”. Arkivendelser vises måske ikke i Finder som en del af arkivnavnet, men du kan stadig bruge endelser til at finde et arkiv i en instruks.

Hvis du bruger kommandoen “info for” til et program, er mappekendetegnet af resultatet normalt “true”. Du kan få flere oplysninger om arkivet og dets navn ved at bruge ekstra kendetegn for resultatet, f.eks. extension hidden, name extension eller displayed name.

BESLÆGTEDE EMNER

path to
info for
Henvis til mapper, arkiver og diske
choose application
Instrukskommandoer til Finder

Finderkommander

Henvis til mapper, arkiver og diske

AppleScript giver dig mulighed for at henvise til visse mapper, arkiver og diske vha. navnet uden en sti. AppleScript kan identificere, hvor sådanne emner ligger, f.eks. i mappen Dokumenter, på skrivebordet, i brugermappen og på startdisken.

- Dette eksempel viser, hvordan du henviser til en speciel mappe vha. dens navn i en instruks.

```
set x to the path to the desktop folder
```

BESLÆGTEDE EMNER

Henvis til Finder i instrukser
path to
info for
Instrukskommandoer til Finder

Finderkommander

Henvis til Finder i instrukser

Hvis du vil bruge Finderkommandoer, sender du kommandoerne til Finder i et afsnit med tell på samme måde, som du sender kommandoer til andre programmer.

- Her er et eksempel på brug af et afsnit med tell i en instruks. Afsnittet med tell henviser funktionen med antallet af vinduer til “Finder.app”.

```
tell application "Finder.app"
        return the number of windows
end tell
```

BESLÆGTEDE EMNER

Finder
Fortæl et objekt, hvad det skal gøre
Henvis til mapper, arkiver og diske
Definitioner af objekter
Instrukskommandoer til Finder

ÅBN

AppleScript Language Guide - Object specifiers

Instrukskommandoer til Finder

Her er nogle kommandoer, du kan bruge med Finder:

activate

alias file

alias list

application file

clean up

close

column

Computer-object

container

copy

count

data size

delete

desktop window

desktop

disk

document file

duplicate

eject

empty trash

exists

file

file information

find

Finder

Finder window

folder

get

icon view options

information window

Internet location file

item

list view options

make

move

moving folder window for

open

package

path to

preferences

preferences window

print

quit

Henvis til mapper, arkiver og diske

Henvis til Finder i instrukser

reveal

select

set

sort

trash

update

window

Finderkommander

Standardkommandoer i instrukstilbehør

Instrukstilbehør tilføjer mange praktiske kommandoer, som du kan inkludere i dine instrukser.

adding folder items to

beep

choose application

choose color

choose file

choose file name

choose folder

choose from list

choose URL

clipboard info

close access

closing folder window for

current date

delay

dialog reply

display

display dialog

do shell script

download

FTP item

get eof

info for

list disks

list folder

load script

mount volume

offset

open for access

open location

opening folder

path to

POSIX path

random number

read

round

run script

say

scripting components

set eof

set the clipboard to

set volume

store script

summarize

system attribute

time to GMT

upload

URL

write